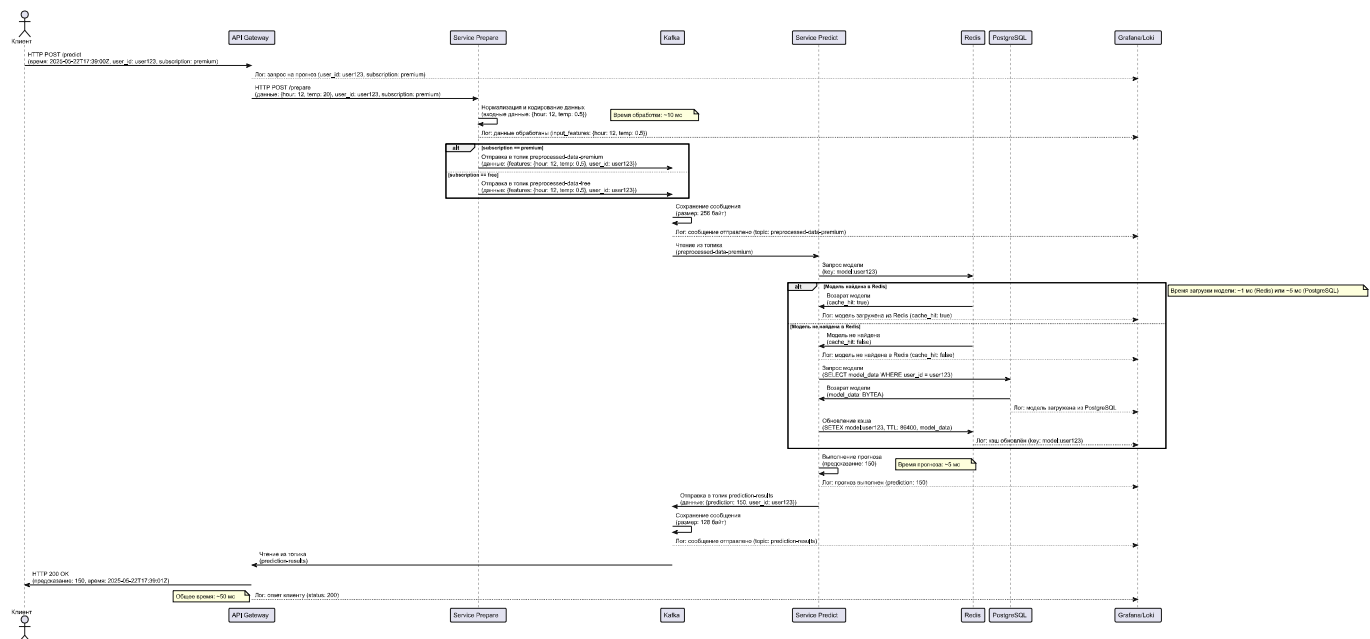


## Процесс прогнозирования

Plantuml source



### 1. Клиент отправляет запрос на прогноз

- <https://wiki.yandex.ru/homepage/257e28581076/process-prognozirovaniya/>

- **Логирование:** PostgreSQL отправляет лог в Grafana/Loki: "модель загружена из PostgreSQL".

#### 8. Service Predict обновляет кэш в Redis

- После загрузки модели из PostgreSQL Service Predict сохраняет её в Redis с ключом `model:user123` и TTL 24 часа (86400 секунд).
- **Логирование:** Redis отправляет лог в Grafana/Loki: `key: model:user123`.

#### 9. Service Predict выполняет прогноз

- Service Predict использует загруженную модель для выполнения прогноза.
- Результат: `prediction: 150`.
- Время прогноза: ~5 мс.
- **Логирование:** Service Predict отправляет лог в Grafana/Loki: `prediction: 150`.

#### 10. Service Predict отправляет результат в Kafka

- Результат прогноза ( `prediction: 150, user_id: user123` ) отправляется в топик `prediction-results`.
- Kafka сохраняет сообщение (размер: 128 байт).
- **Логирование:** Kafka отправляет лог в Grafana/Loki: `topic: prediction-results`.

#### 11. API Gateway считывает результат из Kafka

- API Gateway асинхронно читает сообщение из топика `prediction-results`.
- Получает данные: `prediction: 150, user_id: user123`.
- **Объяснение:** API Gateway ожидает результата через Kafka, что обеспечивает асинхронность и масштабируемость.

#### 12. API Gateway возвращает результат клиенту

- API Gateway отправляет клиенту HTTP-ответ с кодом 200 ОК и результатом: `prediction: 150`.
- Временная метка ответа: `2025-05-22T17:39:01Z`.
- Общее время обработки: ~50 мс.
- **Логирование:** API Gateway отправляет лог в Grafana/Loki: `status: 200`.